IN THE CLAIMS:

The following listing of claims will replace all prior versions, and listings, of claims in the application.

1 - 33. (Canceled)

34. (New) A graphics system comprising:

a graphics processor configured to receive graphics primitives and to perform rendering operations on the graphics primitives to obtain a plurality of samples that represent an image, wherein the rendering operations are performed according to a plurality of screen-space regions, wherein graphics primitives that fall within a first of the regions are rendered with a different density of samples in screen space than graphics primitives that fall within a second of the regions;

a sample buffer coupled to said graphics processor for storing the plurality of samples; and

a sample-to-pixel calculation unit configured to access the samples from the sample buffer and to filter the samples in order to form output pixels.

35. (New) The graphics system as recited in claim 34, wherein the graphics processor is further configured to divide any graphics primitive that falls on a boundary between any two the regions, wherein said dividing divides the primitive into two or more subprimitives, wherein each of the subprimitives falls within exactly one of the two regions.

36. (New) The graphics system as recited in claim 34, wherein the sample-to-pixel calculation unit is configured to filter the samples using an area of support that varies between at least two of said regions.

37. (New) The graphics system as recited in claim 34, wherein each of the regions has a different assigned sample density, wherein the graphics processor is configured so that the regions are movable in screen space.

38. (New) The graphics system as recited in claim 37, wherein the graphics processor is configured to move one or more of the regions based on input from one or more of the following: an eye-tracking device, a head-tracking device, a hand-tracking device, a mouse, a keyboard.

39. (New) The graphics system as recited in claim 37, wherein the graphics processor is configured to move one or more of the regions based on input defining a current point of interest.

40. (New) The graphics system as recited in claim 39, wherein the current point of interest is determined by a current position of a cursor, a current position of a main character represented within the graphics primitives, or a current position of a visible object represented within the graphics primitives.

41. (New) The graphics system as recited in claim 34, wherein each of the regions has a different assigned sample density, wherein the graphics processor is configured to move at least a first of the regions having the highest sample density, wherein the graphics processor is configured to center the first region on a current point of foveation.

42. (New) The graphics system as recited in claim 34, wherein the graphics processor is configured to receive input that dynamically tracks the current point of foveation.

43. (New) The graphics system as recited in claim 34, wherein each of the regions is assigned a different assigned sample density, wherein the assignment of sample densities to regions is based on an edge analysis.

44. (New) The graphics system as recited in claim 34, wherein the graphics processor is further configured to vary the density of the samples that are generated within at least one of the regions on a basis selected from the group consisting of: a per-scan line basis, a per-group-of-scan-lines basis, a per-pixel basis, and a per-group-of-pixels basis.

45. (New) The graphics system as recited in claim 34, the graphics system is configured to move the regions according to input from a gaze tracking device.

46. (New) The graphics system as recited in claim 34, wherein the density of samples is substantially continuously variable across one or more boundaries between regions.

47. (New) The graphics system as recited in claim 34, further comprising a sample position memory configured to store information usable to determine sample positions at a number of different densities in screen space, wherein the graphics processor is configured to receive the sample positions from said sample position memory according to a selected sample density.

48. (New) The graphics system as recited in claim 1, wherein the sample buffer is configured to double buffer at least the color components of said samples.

49. (New) A method for producing output pixels for a graphics system, the method comprising:

receiving graphics primitives;

performing rendering operations on the graphics primitives to obtain a plurality of samples that represent an image, wherein the rendering operations are performed according to a plurality of screen-space regions, wherein graphics primitives that fall within a first of the regions are rendered with a different density of samples in screen space than graphics primitives that fall within a second of the regions;

storing the plurality of samples in a sample buffer, wherein the stored plurality of samples corresponds to the image; and

accessing the samples from the sample buffer and filtering the samples to form the output pixels, wherein the output pixels correspond to a displayable frame.

50. (New) The method as recited in claim 49, wherein each of the regions has a different assigned sample density, the method further comprising:

moving one or more of the regions in screen space in response to dynamic input specifying a point of interest.

51. (New) The method as recited in claim 49 further comprising:

divide a first of the graphics primitive that falls on a boundary between two of the regions, wherein said dividing divides the first primitive into two or more subprimitives, wherein each of the subprimitives falls within exactly one of the two regions.

52. (New) A graphics system comprising:

a graphics processor configured to receive graphics primitives and render the graphics primitives to obtain a plurality of samples that represent a frame, wherein said rendering is performed according to a plurality of screen-space regions, wherein graphics primitives that fall within a first of the regions are rendered with a different density of samples in screen space than graphics primitives that fall within a second of the regions;

a sample buffer coupled to said graphics processor for storing the samples, wherein the samples are stored in bins that corresponds to equal areas in screen space, wherein the number of samples stored per bin in the sample buffer varies according to said regions;

a convolution unit configured to access the samples from the sample buffer and to filter the samples in order to form output pixels.

53. (New) The graphics system as recited in claim 52, wherein the graphics processor is further configured to divide any graphics primitive that falls on a boundary between any two the regions, wherein said dividing divides the primitive into two or more subprimitives, wherein each of the subprimitives falls within exactly one of the two regions.

54. (New) The graphics system as recited in claim 52, wherein the convolution unit is configured to filter the samples using an area of influence that varies between at least two of said regions.

55. (New) The graphics system of claim 52, wherein the graphics processor is configured to move the regions based on input from a tracking device.